

# **A dynamic and generic *cloud* computing model for environmental analysis using *in-situ* sensing data applied to glacier mass balance**

**(Mer de Glace, Chamonix;  
East Loven Glacier, Spitsberg)**

**E. RANISAVLJEVIC<sup>1</sup>, F. DEVIN<sup>1</sup>, Y. LE NIR<sup>1</sup>,  
J.M. FRIEDT<sup>2</sup>, C. MARLIN<sup>3</sup>, M. GRISELIN<sup>4</sup>, D. LAFFLY<sup>5</sup>**

<sup>1</sup> Ecole Internationale des Sciences du Traitement de l'Information, Pau, France

<sup>2</sup> Univ. de Franche-Comté, FEMTO-ST, UMR 6174 CNRS, Besançon, France

<sup>3</sup> Univ. de Paris-sud, IDES, UMR 8148 CNRS, Orsay, France

<sup>4</sup> Univ. de Franche-Comté, THEMA, UMR 6049 CNRS, Besançon, France

<sup>5</sup> Univ. de Toulouse, GEODE, UMR 5602 CNRS, Toulouse, France



# Summary

## Introduction

- I. Snow cover and glacier evolution
- II. In-situ sensing constraints
- III. Cloud computing and Web Services
- IV. Tasks architecture
- V. Model

## Conclusion

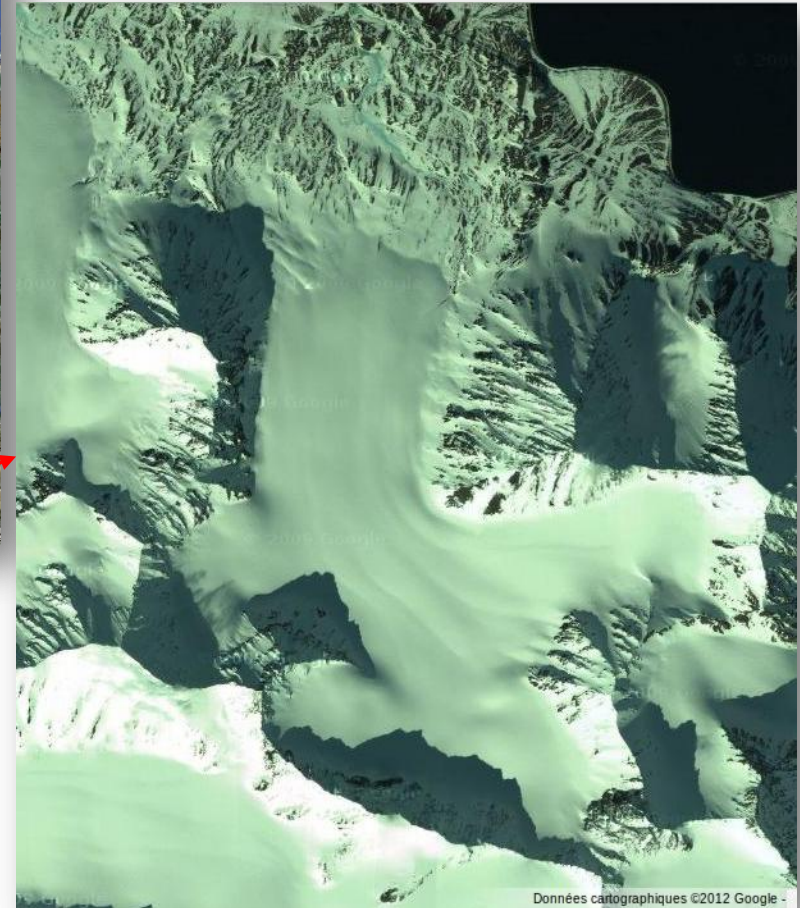
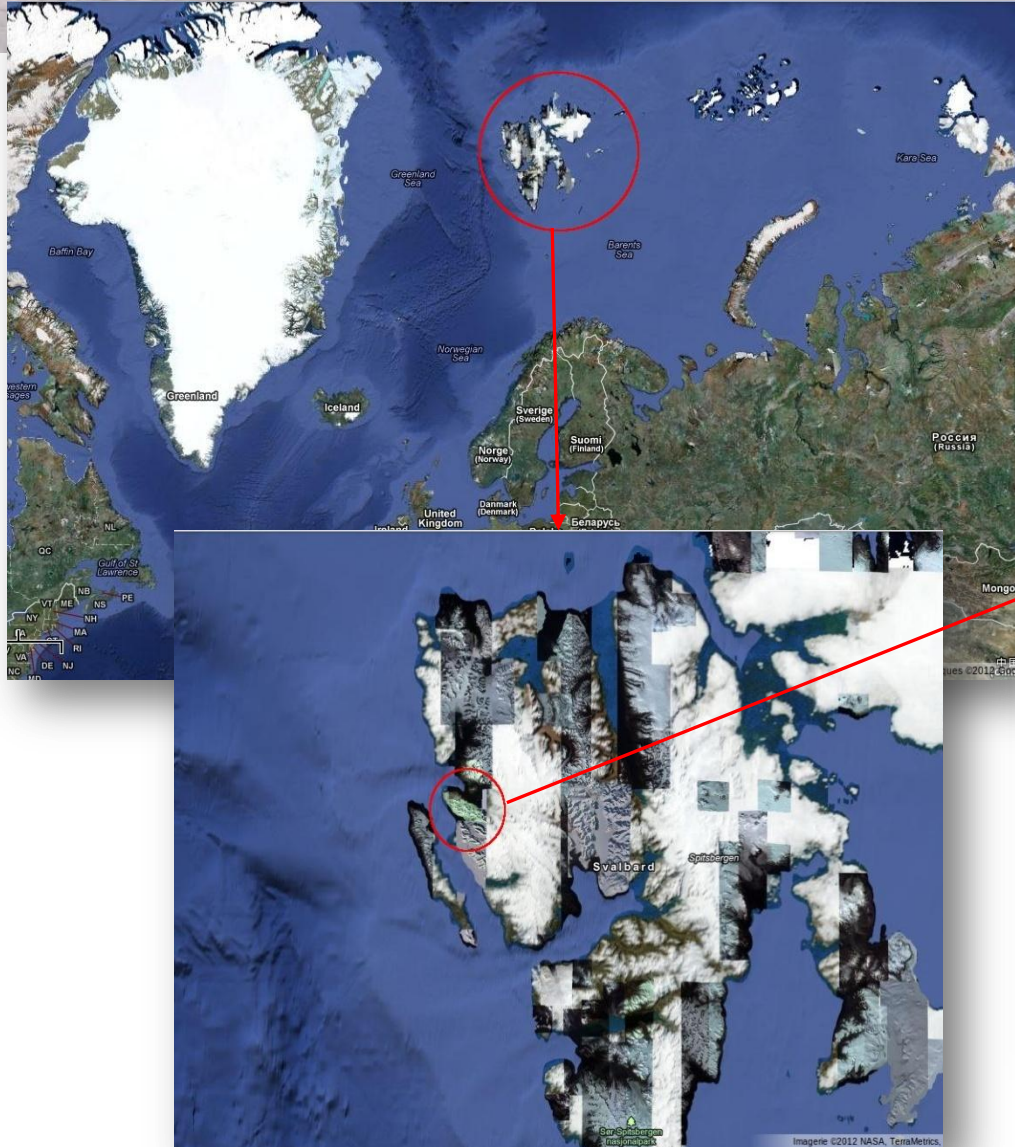
# Introduction

- Hydro Sensor FLOWS project, supervised by M. Griselin (C. Marlin and D. Laffly)
  - Map the temporal evolution of the snow cover
  - Couple it with a hydrologic model
- East Loven glacier: experimental field
  - 4 years of different readings
  - Generate a huge data base

How can Cloud Computing improve the processing of the data base ?



# East Loven glacier



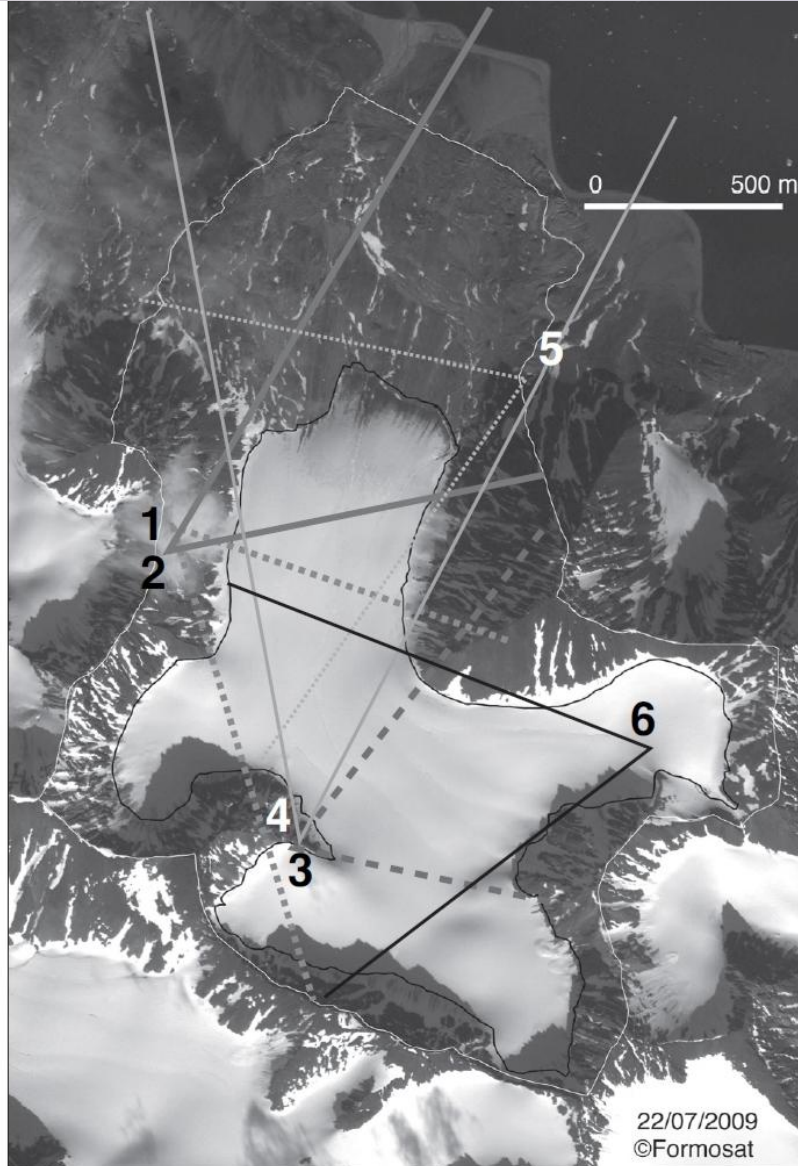
# Snow Cover and Glacier Evolution

- **Remote sensing:** daily satellite imagery is not always accessible
  - Cost
  - Poor weather conditions (heavy cloud cover)
  - Fast events not visible
- ***In situ* sensing:** Ground based autonomous automated digital camera
  - 3 pictures / day
  - Huge data base
  - Reconstruct the satellite view

(D. Laffly et *al.*, Cambridge 2010)



# Six digital cameras are positioned around the glacier basin, providing complete glacier coverage

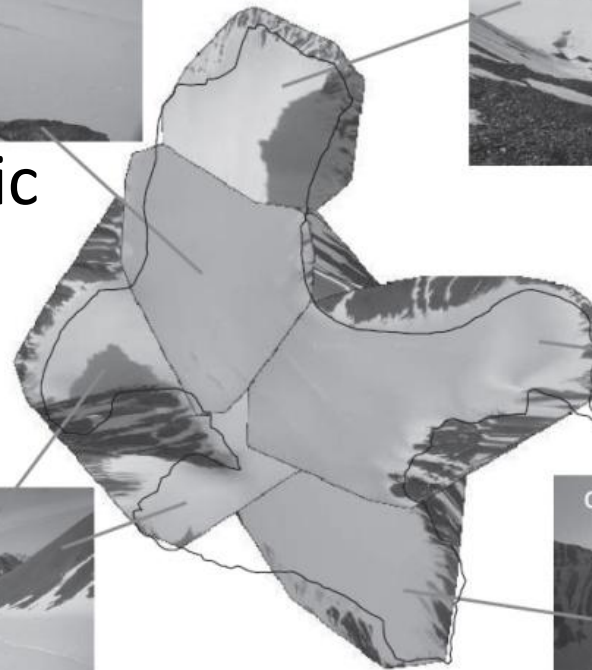


# *In-situ* sensing: image processing

- Projection of the picture, from the oblique view to a plan view



- Construction of a mosaic
- Classify the different phenomena (snow, ice)
- Constraints processing



07/22/2009

# *In-situ* Constraints: Atmospheric Disturbance





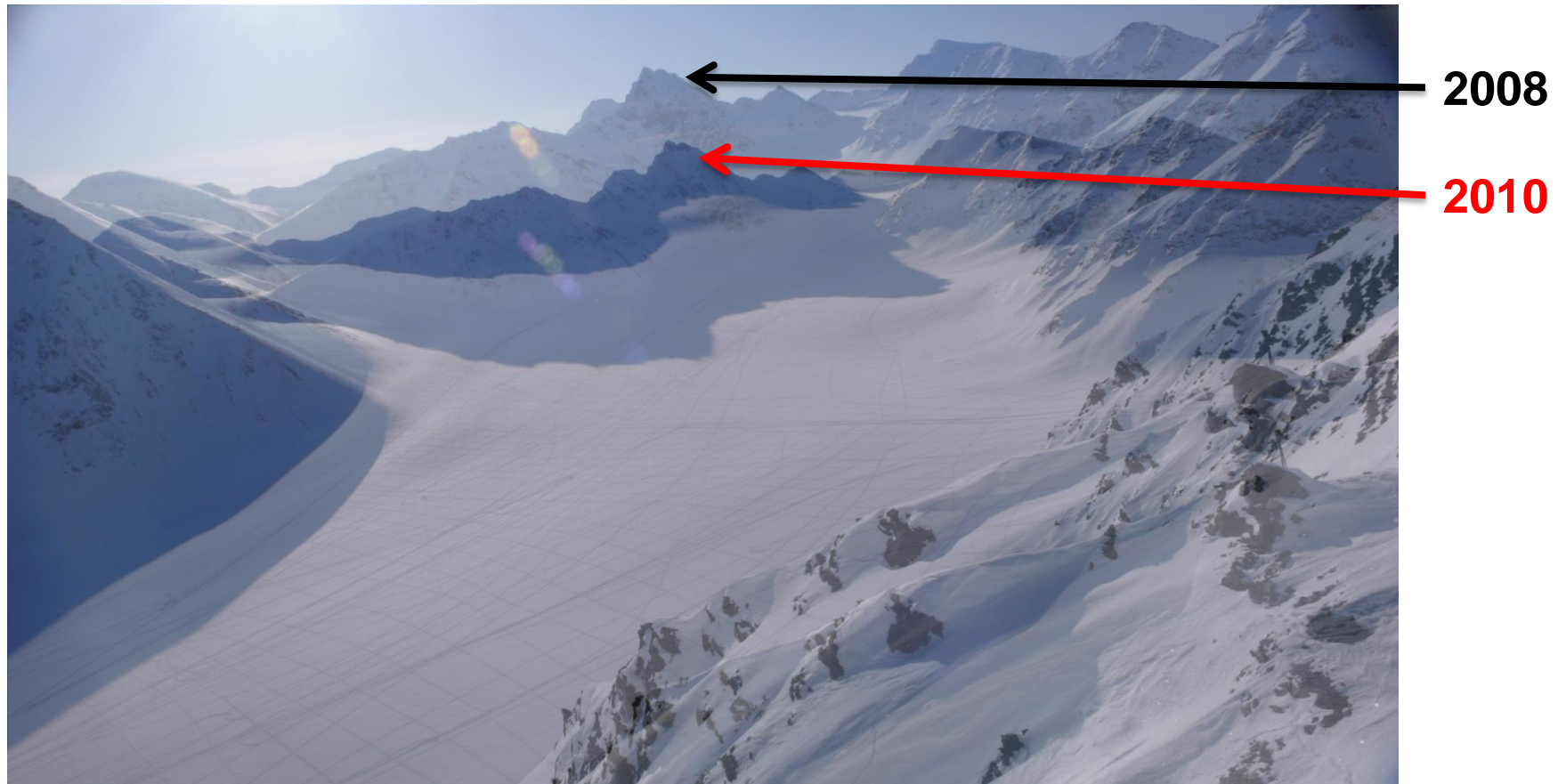
# *In-situ* Constraints: Electronic Deficiency



- Poor weather conditions
  - Microcontroller “asleep” for several days
- Electromagnetic perturbation: reset of the microcontroller
  - 6 pictures / day
- Discharge of the camera intern battery
  - Loss of the picture’s date

# *In-situ* Constraints: Geometry Variations

- Modification of the shooting's parameters



# *In-situ* constraints adjusted by computer

- Manually process all the pictures  
(over 30 000)
- Specific and heterogenous tools

Real need to provide an application :

- dynamically change the processing
- be as generic as possible to fit other needs
- avoid human operations as much as possible



# Cloud Computing

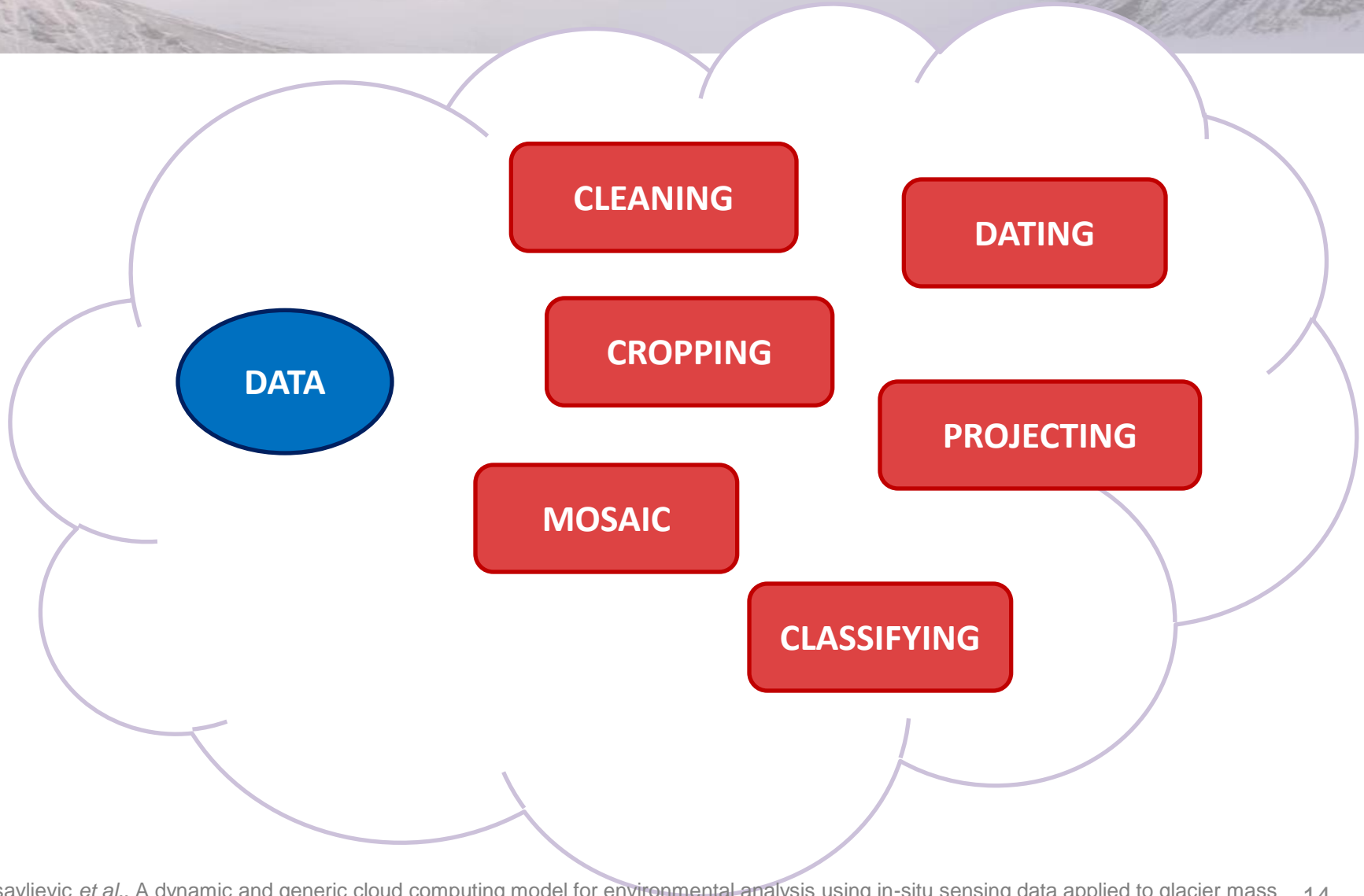
Set of resources, servers and applications offered “as a service” over a network.

- Advantages:
  - Easiness of access
  - Large storage capacity
  - Lightness of application
  - Modular : add, remove, modify services
  - Scalable
    - Increase of users connected to the service
    - Increase the computing capacity according to the needs
    - Fault tolerance

# Web Services

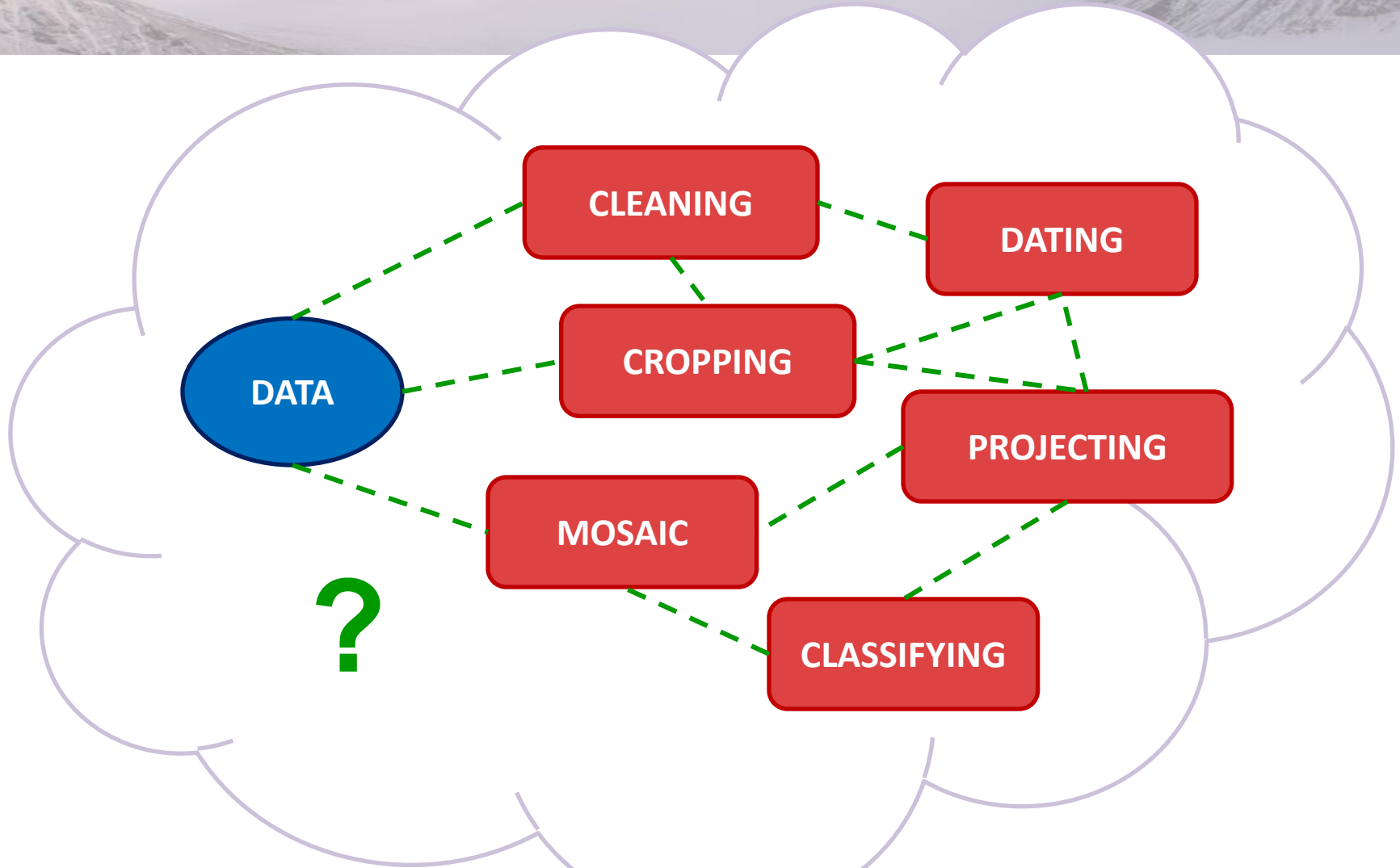
- Software offered as a service (SAAS)
- Multiple tasks
  - Cleaning / Usability
  - Dating
  - Cropping
  - Projecting
  - Constructing a mosaic
  - Classify glacier's phenomena (ice, snow ...)

# Tasks Architecture

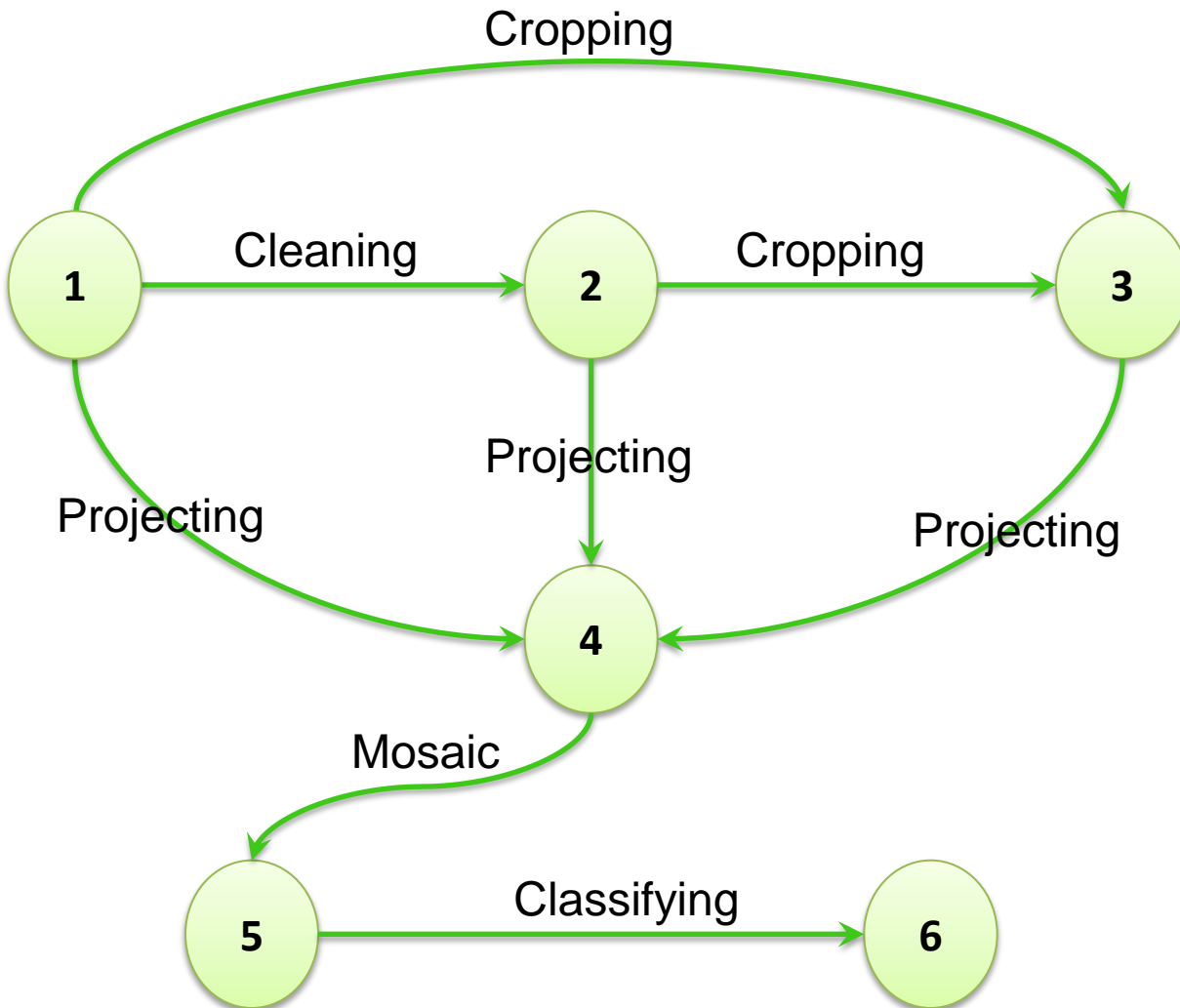




# Tasks Architecture



# Planner and Scheduler



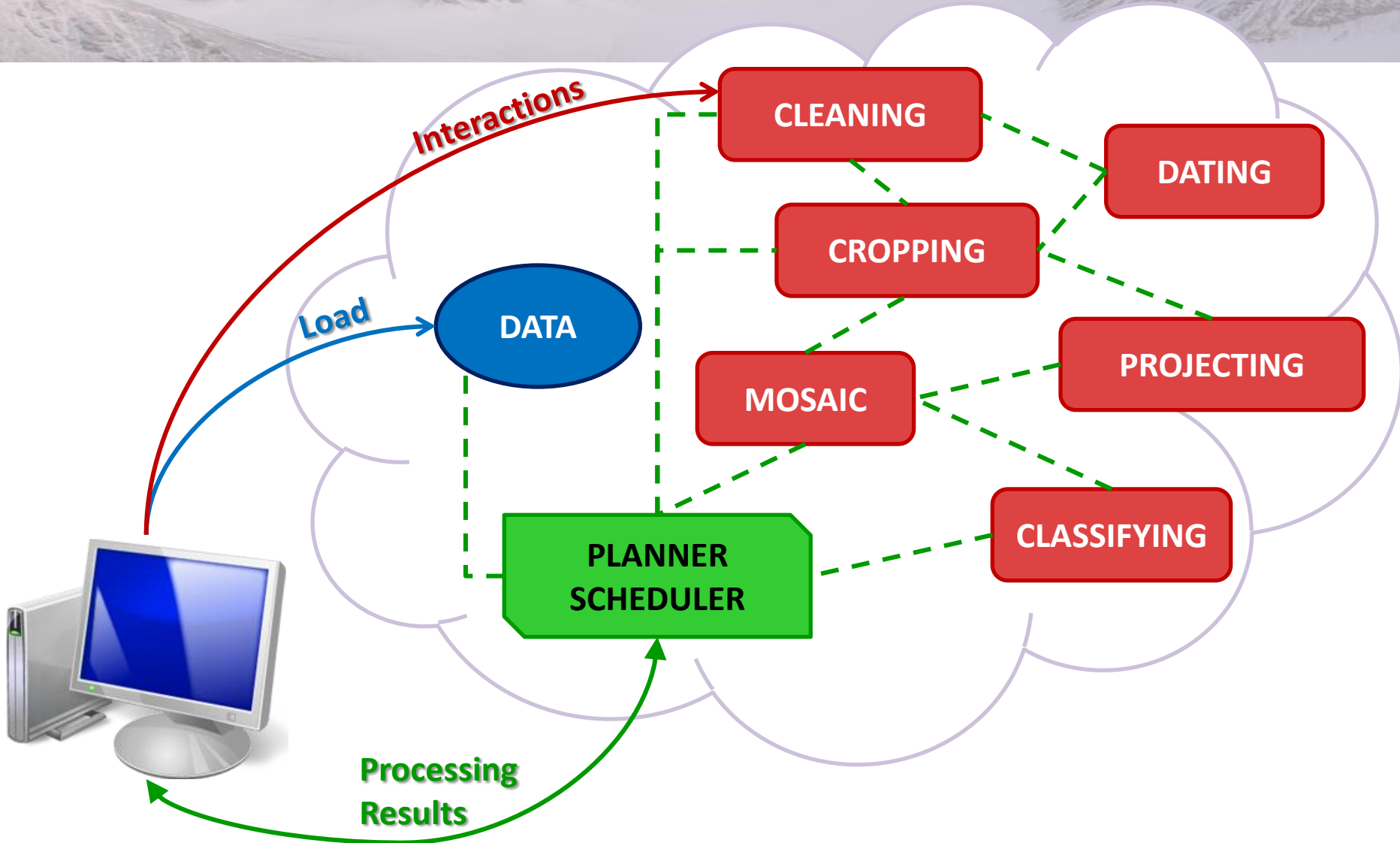
## Planner:

- define a workflow
- achieve a goal
- satisfy constraints

## Scheduler:

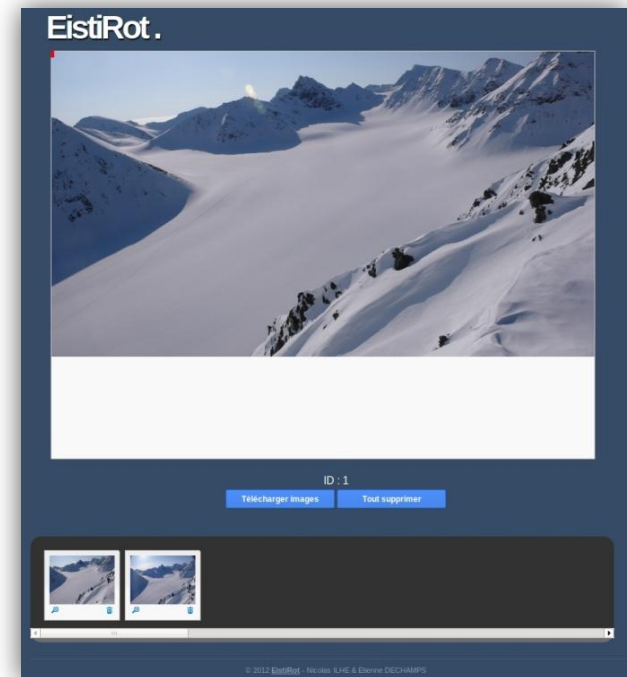
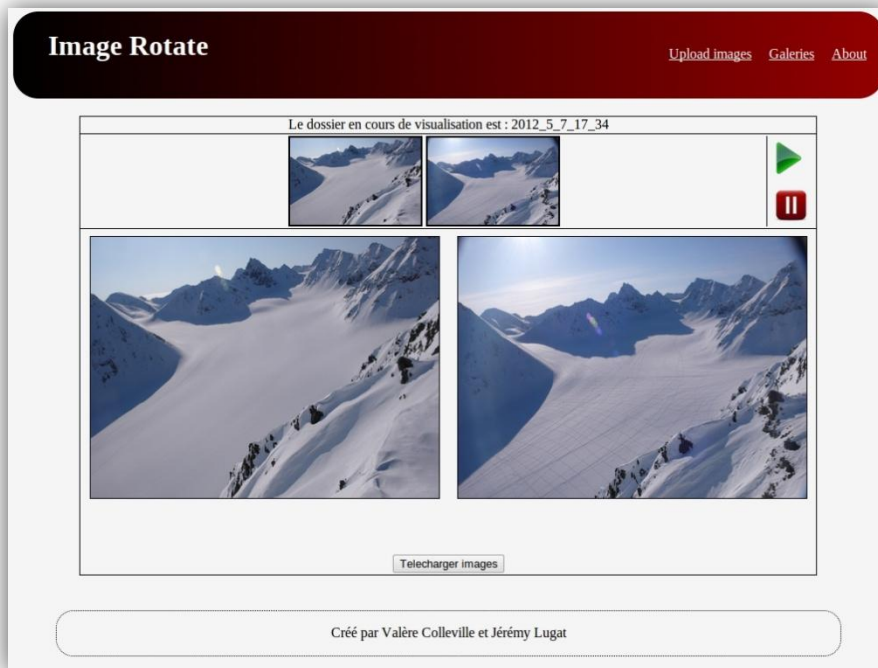
- organize workflow
- temporal constraints

# Model





# Example : Cropping Service



# Conclusion

- Model :
  - Generic (web services)
  - Dynamic (planner and scheduler)
- The environmental sciences and the geoengineering generate a huge data base (Big Data). Cloud computing is an answer to some processing constraints and storage constraints.
- Evolution of the languages and paradigms of computer science.

# Thank you

